UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/813,003	03/31/2004	Jerry Chow	NRT.0199US (15392ROUS04U)	5213

21906 7590 02/05/2010

TROP, PRUNER & HU, P.C.
1616 S. VOSS ROAD, SUITE 750
HOUSTON, TX 77057-2631

EXAMINER	
KIM, JUNG W	

ART UNIT	PAPER NUMBER
2432	

MAIL DATE	DELIVERY MODE
02/05/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/813,003
Filing Date: March 31, 2004
Appellant(s): CHOW, JERRY

Dan C. Hu
Registration No. 40,025
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 11/2/09 appealing from the Office action mailed 5/20/09.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is substantially correct. A new ground of rejection is made for claims 26-28.

NEW GROUND(S) OF REJECTION

Claims 26-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hind et al. US 6,976,163 in view of Bryant US 5,628,023 and Bishop Computer Security, Chapter 29.5 "Common Security-Related Programming Problems." Dependent claims 26-28 were originally rejected in the final rejection mailed on 5/20/09 as if dependent on independent claim 22. This was an oversight as these dependent

claims are actually dependent on claim 44. The new grounds of rejection rectify this oversight. The rejections now identify claims 26-28 as being dependent on claim 44. Claim 44 in turn is a dependent claim of claim 25, which in turn is dependent on claim 22.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

5628023	BRYANT ET AL.	5-1997
20020124148	BEUKEMA ET AL.	9-2002
6976163	HIND ET AL.	12-2005
7194092	ENGLAND ET AL.	3-2007

Bishop, Matt "Computer Security," December 12, 2002, Chapter 29.5, "Common Security-Related Programming Problems" pp. 887-913.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1, 9, 10, 22-25, 30, 32, 34, 35, 41 and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bryant et al. US 5,628,023 (hereinafter Bryant) in view of Bishop Computer Security, Chapter 29.5 "Common Security-Related Programming Problems" (hereinafter Bishop).

As per claims 1, 9, 10, 41 and 46, Bryant discloses a memory protection system comprising:

a key store to store identifiers of protected memory locations and respective corresponding memory protection keys; and a memory access manager including at least hardware configured to receive a memory command for altering contents of any of the protected memory locations, determine whether the memory command includes a memory protection key corresponding to at least one of said protected memory locations to be altered wherein the memory protection key in the memory command is written to a volatile memory, if the memory command includes the memory protection key corresponding to each protected memory location to be altered, permit the memory command to proceed, and then render the memory protection key in the memory command inaccessible (col. 5:55-6:20, the token is accessed *from* the register to *present* the token to the protection verification process; fig. 3, reference nos. 485-530 and 535-560; figs. 7 and 9; by virtue of de-allocating memory; see also applicant's specification, pg. 17, lines 19-26);

implemented in an electronic device having a memory, the memory comprising the protected memory locations and unprotected memory locations (program requests region of memory to be protected);

wherein the memory access manager is further configured to receive memory commands for altering contents of the unprotected memory locations

without checking for any memory protection key (only protected memory is verified [see fig. 3]);

wherein the memory access manager is configured to further receive a memory read command to read content of a particular protected memory location, the memory access manager to allow the memory read command to proceed to read the content of the particular protected memory location without checking for any memory protection key. (col. 21:3-22)

Although Bryant does not disclose rendering the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory protection key written to the volatile storage such that the memory protection key written to the volatile memory is inaccessible after completion of the memory command, the step of erasing sensitive information to prevent unauthorized disclosure of protected information is well known in the art. Such a step prevents covert analysis of memory to determine the value of deallocated memory. For example, Bishop discloses a basic tenet of secure deletion of sensitive information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence, emphasis added) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made to render the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory

protection key written to the volatile storage such that the memory protection key written to the volatile memory is inaccessible after completion of the memory command. One would be motivated to do so to securely remove sensitive information as known to one of ordinary skill. The aforementioned cover the limitations of claims 1, 9, 10, 41 and 46.

As per claims 22-25, 30, 32, 34 and 35, Bryant discloses a method of protecting memory in an electronic device, comprising:

receiving a memory command to access a protected memory location; determining whether the received memory command is a memory read command to read the protected memory location, or a memory write command to alter the protected memory location (col. 21:3-21; fig. 3); in response to determining that the received memory command is the memory write command:

identifying a memory protection key corresponding to the protected memory location; determining whether the memory command includes the memory protection key corresponding to the protected memory location, wherein at least the memory protection key in the memory write command has been written to volatile memory; permitting completion of the memory write command if the memory command includes the memory protection key corresponding to the protected memory location (col. 5:55-6:20, the token is accessed *from* the register to *present* the token to the protection verification process; fig. 3, reference nos. 540, 560); and rendering the memory protection key in the memory command that has been written to

the volatile memory inaccessible (by virtue of de-allocating memory; see also applicant's specification, pg. 17, lines 19-26); and

in response to determining that the received memory command is the memory read command, processing the memory read command to read the protected memory location without checking for any memory protection key; (21:15-18)

wherein permitting comprises performing the memory write command (fig. 3, reference no. 570);

wherein receiving comprises receiving the memory command from an originating electronic device component, and wherein permitting comprises allowing the originating electronic device component to perform the memory write command; (fig. 1, reference nos. 100, 105, 110)

receiving data to be written to the protected memory location; and generating the memory write command responsive to receiving the data (fig. 3, reference no. 540);

wherein identifying comprises identifying a protected memory location in the memory write command and accessing a mapping table that maps protected memory locations to respective corresponding memory protection keys (fig. 1, reference nos. 140, 145, 155, 175 and 185);

further comprising: receiving memory commands to alter unprotected memory locations; and permitting completion of the memory commands to alter

unprotected memory locations without checking for any memory protection keys (unprotected memory does not require verification);

wherein the identifying step comprises accessing the memory protection key corresponding to the protected memory location in a key store, the method further comprising:

receiving a command to establish a new protected memory location in the memory and a memory protection key corresponding to the new protected memory location; establishing the new protected memory location in the memory; and storing the memory protection key in the key store. (fig. 3, reference nos. 485-530; figs. 7 and 9)

Bryant further discloses a computer-readable medium storing instructions for performing the method of claim 22. (fig. 1)

Although Bryant does not disclose rendering the memory protection key in the memory command that has been written to the volatile memory inaccessible by overwriting at least a portion of the memory protection key in the volatile memory upon completion of the memory write command to make the memory protection key in the volatile memory inaccessible after completion of the memory write command, the step of erasing sensitive information to prevent unauthorized disclosure protected information is well known in the art. Such a step prevents covert analysis of memory to determine the value of deallocated memory. For example, Bishop discloses a basic tenet of secure deletion of sensitive information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object

should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence, emphasis added) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made to render the memory protection key in the memory command that has been written to the volatile memory inaccessible by overwriting at least a portion of the memory protection key in the volatile memory upon completion of the memory write command to make the memory protection key in the volatile memory inaccessible after completion of the memory write command. One would be motivated to do so to securely remove sensitive information as known to one of ordinary skill. The aforementioned cover the limitations of claims 22-25, 30, 32, 34 and 35.

Claims 1, 2, 4, 7-9, 11-15 and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Beukema et al. US Patent Application Publication No. 20020124148 (hereinafter Beukema) in view of Bishop.

As per claims 1, 2, 4, 7-9, 11-15 and 46, Beukema discloses a memory protection system comprising:

a key store to store identifiers of protected memory locations and respective corresponding memory protection keys; and a memory access manager including at least hardware configured to receive a memory command

for altering contents of any of the protected memory locations, determine whether the memory command includes a memory protection key corresponding to at least one of said protected memory locations to be altered, wherein the memory protection key in the memory command is written to a volatile memory, if the memory command includes the memory protection key corresponding to each protected memory location to be altered, permit the memory command to proceed, and then render the memory protection key in the memory command inaccessible (paragraph 54; by virtue of de-allocating memory; see also applicant's specification, pg. 17, lines 19-26);

wherein the identifiers comprise addresses in a protected memory; wherein the identifiers identify data entries in a protected memory; (paragraph 54; pointer to an associated memory region/address)

wherein the key store stores a mapping table that maps each identifier to a corresponding memory protection key; (paragraph 54; "Protection/Translation Table");

wherein at least one of the identifiers is mapped to multiple corresponding memory protection keys (paragraph 54; L_key and R_key);

the system implemented in an electronic device having a memory, the memory comprising the protected memory locations and unprotected memory locations (paragraph 55, and fig. 6);

wherein the memory access manager is further configured to perform the memory command that includes the memory protection key corresponding to each protected memory location to be altered (paragraphs 54 and 59);

the system implemented in an electronic device, wherein the memory command is received by the memory access manager from an originating electronic device component, and wherein the originating electronic device component proceeds with the memory command permitted by the memory access manager; wherein the originating electronic device component is a memory update module; wherein the originating electronic device component sends memory commands to the memory access manager responsive to data received at the electronic device; wherein the originating electronic device component is further configured to extract a received memory protection key from the received data and to provide the received memory protection key to the memory access manager. (fig. 2; paragraphs 54-56; external user supplies protection key for rights access (read, write) to protected memory)

Although Beukema does not disclose rendering the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory protection key written to the volatile storage such that the memory protection key written to the volatile memory is inaccessible after completion of the memory write command, the step of erasing sensitive information to prevent unauthorized disclosure protected information is well known in the art. Such a step prevents covert analysis of memory to determine the value of deallocated memory. For example, Bishop discloses a basic

tenet of secure deletion of sensitive information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence, emphasis added) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made to render the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory protection key written to the volatile memory is inaccessible after completion of the memory write command. One would be motivated to do so to securely remove sensitive information as known to one of ordinary skill. The aforementioned cover the limitations of claims 1, 2, 4, 7-9, 11-15 and 46.

New grounds of Rejection:

Claims 16, 17, 19, 20, 22, 25-28, 36, 39, 43-45 and 47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hind et al. US 6,976,163 (hereinafter Hind) in view of Bryant and Bishop.

As per claims 16, 17, 19, 20 and 43, Hind discloses an electronic device comprising a memory; a wireless receiver configured to receive data relating to a remote software update to be written to the memory, and means to securely update the software files via update rules. (col. 2:38-59; 19:40-46) However, Hind does not disclose

ensuring that the update has the proper permission to execute the update in a particular memory region. Bryant discloses an electronic device comprising:

a memory; a receiver configured to receive data to be written to the memory; and a memory protection system associating protected memory locations in the memory with respective corresponding keys, and configured to allow the received data to be written to any of the protected memory locations only if the received data includes a key corresponding to the protected memory location to which the received data is to be written and to render the corresponding key in the received data inaccessible after allowing the received data to be written to the protected memory location (fig. 1, fig. 3);

volatile storage having unprotected memory locations, the memory protection system configured to download the received data including the key to the unprotected memory locations of the volatile storage prior to writing the received data to the protected memory locations; wherein the volatile storage is part of the memory (fig. 1, reference nos. 125, 130 and 140);

wherein each key is rendered inaccessible by erasing the received data from the unprotected memory locations where the memory access manager allows the received data to be written to the protected memory locations (by virtue of de-allocating memory; see also applicant's specification, pg. 17, lines 19-26);

wherein the memory protection system comprises: a key store storing a mapping table that associates the protected memory locations with the

respective corresponding keys; and a memory access manager configured to process a memory command for writing the received data to any of the protected memory locations, determine whether the received data includes the key corresponding to any of the protected memory locations to which the received data is to be written, if the received data includes the key corresponding to a protected memory location to which the received data is to be written, to permit the memory command to proceed, and then render the corresponding key in the received data inaccessible (19:41-20:6);

wherein the key store resides at a secure location in the memory outside of the main memory (fig. 1, reference no. 105);

wherein the memory protection system is configured to further receive a memory read command to access a particular one of the protected memory locations, perform reading of the particular protected memory location in response to the memory read command, without checking for any memory protection key. (21:3-22)

It would be obvious to one of ordinary skill in the art at the time the invention was made to modify the invention of Hind with the teaching of Bryant. One would be motivated to do so to ensure that the update has the proper permission to execute the update in a particular memory region as disclosed by Bryant. (5:22-30)

Furthermore, although Bryant does not disclose the memory protection system to render the key inaccessible by overwriting at least a portion of the key, the step of erasing sensitive information to prevent unauthorized disclosure protected information is

well known in the art. Such a step prevents covert analysis of memory to determine the value of deallocated memory. For example, Bishop discloses a basic tenet of secure deletion of sensitive information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence, emphasis added) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made for the memory protection system to render the key inaccessible by overwriting at least a portion of the key. One would be motivated to do so to securely remove sensitive information as known to one of ordinary skill. The aforementioned cover the limitations of claims 16, 17, 19, 20 and 43.

As per claims 22, 25-27 and 44, Hind discloses a method to remotely update software via update rules contained in the update; receiving the update comprises receiving, by a wireless receiver. (col. 2:38-59; 19:40-46) However, Hind does not disclose ensuring that the update has the proper permission to execute the update in a particular memory region. Bryant discloses a method of protecting memory in an electronic device, comprising:

receiving a memory command to access a protected memory location;

determining whether the received memory command is a memory read command to read the protected memory location, or a memory write command to

alter the protected memory location (col. 21:3-21; fig. 3); in response to determining that the received memory command is the memory write command:

identifying a memory protection key corresponding to the protected memory location; determining whether the memory command includes the memory protection key corresponding to the protected memory location, wherein at least the memory protection key in the memory write command has been written to volatile memory; permitting completion of the memory write command if the memory write command includes the memory protection key corresponding to the protected memory location (col. 5:55-6:20, the token is accessed *from* the register to *present* the token to the protection verification process; fig. 3, reference nos. 540, 560); and rendering the memory protection key in the memory command that has been written to the volatile memory inaccessible (by virtue of de-allocating memory; see also applicant's specification, pg. 17, lines 19-26); and

in response to determining that the received memory command is the memory read command, processing the memory read command to read the protected memory location without checking for any memory protection key; (21:15-18)

receiving data to be written to the protected memory location; and generating the memory write command responsive to receiving the data (fig. 1, reference no. 540);

wherein the received data comprises a received key, and wherein generating comprises extracting the received key from the received data and inserting the received key into the memory write command (fig. 3, reference nos. 540 and 550);

wherein determining comprises comparing the memory protection key corresponding to the protected memory location with the received key in the memory write command (fig. 3, reference nos. 555 and 560).

It would be obvious to one of ordinary skill in the art at the time the invention was made to modify the invention of Hind with the teaching of Bryant. One would be motivated to do so to ensure that the update has the proper permission to execute the update in a particular memory region as disclosed by Bryant. (5:22-30)

Finally, although Bryant does not disclose rendering the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory protection key in the volatile memory upon completion of the memory write command to make the memory protection key in the volatile memory inaccessible after completion of the memory write command, the step of erasing sensitive information to prevent unauthorized disclosure protected information is well known in the art. Such a step prevents covert analysis of memory to determine the value of deallocated memory. For example, Bishop discloses a basic tenet of secure deletion of sensitive information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901,

last sentence-pg. 902, first sentence, emphasis added) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made to render the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory protection key in the volatile memory upon completion of the memory write command to make the memory protection key in the volatile memory inaccessible after completion of the memory write command. One would be motivated to do so to securely remove sensitive information as known to one of ordinary skill. The aforementioned cover the limitations of claims 22, 25-27 and 44.

As per claim 28, the rejection of claim 26 under 35 USC 103(a) as being unpatentable over Hind in view of Bryant and Bishop is incorporated herein. Neither Hind, Bryant nor Bishop expressly disclose wherein determining comprises retrieving a modified version of the memory protection key corresponding to the protected memory location, modifying the received key in the memory write command to generate a modified received key, and comparing the modified received key to the modified version of the memory protection key corresponding to the protected memory location. However, it is notoriously well known in the art to use and store a hash value of an identifier as opposed to the original identifier. A hash value uniquely maps an original value to a modified value, such that the modified value is typically much smaller than the original value. Hence, the modified value retains the unique property of the original value but

requires less memory and bandwidth requirements to store and communicate the value. Official Notice of this teaching is taken. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made wherein the determining step comprises retrieving a modified version of the memory protection key corresponding to the protected memory location, modifying the received key in the memory write command to generate a modified received key, and comparing the modified received key to the modified version of the memory protection key corresponding to the protected memory location. One would be motivated to do so to preserve memory and processing resources as known to one of ordinary skill in the art. The aforementioned cover the limitations of claim 28.

As per claims 36, 45 and 47, Hind discloses a method to remotely update software via update rules contained in the update; wherein the update is received by a wireless receiver. (col. 2:38-59; 19:40-46) However, Hind does not disclose ensuring that the update has the proper permission to execute the update in a particular memory region. Bryant discloses a method of protecting electronic memory, comprising:

> configuring a memory store of an electronic device into at least one protected memory location and a key store operable to store an identifier of each protected memory location and a respective corresponding memory protection key; and configuring a processor of the electronic device to provide a memory access manager operable to receive memory commands for altering contents of any of the at least one protected memory location, and for at least one memory

command, to :determine whether the at least one memory command includes a memory protection key corresponding to at least one protected memory location to be modified, said memory command including the memory protection key corresponding to at least one said protected memory location to be modified, permit the at least one memory command and then render each corresponding memory protection key in the at least one memory command inaccessible; wherein the memory protection key in the at least one memory command is written to volatile memory, and wherein the memory protection key in the at least one memory command is rendered inaccessible (col. 5:55-6:20, the token is accessed *from* the register to *present* the token to the protection verification process; fig. 3, reference nos. 540-570; by virtue of de-allocating memory; see also applicant's specification, pg. 17, lines 19-26)

wherein configuring the processor further comprises configuring the processor to receive a memory read command to read a particular one of the protected memory locations, and to permit the memory read command to read the particular protected memory location without checking for any memory protection key. (21:3-22)

It would be obvious to one of ordinary skill in the art at the time the invention was made to modify the invention of Hind with the teaching of Bryant. One would be motivated to do so to ensure that the update has the proper permission to execute the update in a particular memory region as disclosed by Bryant. (5:22-30)

Finally, although Bryant does not disclose rendering the memory protection key in the at least one memory command inaccessible by overwriting at least a portion of the memory protection key written to the volatile memory such that the memory protection key written to the volatile memory is rendered inaccessible after completion of the at least one memory command, the step of erasing sensitive information to prevent unauthorized disclosure protected information is well known in the art. Such a step prevents covert analysis of memory to determine the value of deallocated memory. For example, Bishop discloses a basic tenet of secure deletion of sensitive information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence, emphasis added) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made to render the memory protection key in the at least one memory command inaccessible by overwriting at least a portion of the memory protection key written to the volatile memory such that the memory protection key written to the volatile memory is rendered inaccessible after completion of the at least one memory command. One would be motivated to do so to securely remove sensitive information as known to one of ordinary skill. The aforementioned cover the limitations of claims 36, 45 and 47.

As per claim 39, Bryant further suggests a computer-readable medium storing instructions for performing the method of claim 36. (fig. 1)

Claims 1 and 3-6 are rejected under 35 U.S.C. 103(a) as being unpatentable over England et al. USPN 7,194,092 (hereinafter England) in view of Bishop.

As per claims 1 and 3-6, England discloses a memory protection system comprising:

> a key store to store identifiers of protected memory locations and respective corresponding memory protection keys; and a memory access manager including at least hardware configured to receive a memory command for altering contents of any of the protected memory locations, determine whether the memory command includes a memory protection key corresponding to at least one of said protected memory locations to be altered, wherein the memory protection key in the memory command is written to a volatile memory, if the memory command includes the memory protection key corresponding to each protected memory location to be altered, permit the memory command to proceed, and then render the memory protection key in the memory command inaccessible (col. 10:41-51, when an application wants to access protected stored content, the application passes its rights manager certificate and storage key to the DRMOS, storage of the storage key in volatile memory is inherent in this step; by virtue of de-allocating memory; see also applicant's specification, pg. 17, lines 19-26);

wherein the identifiers comprise names of protected files in a memory; wherein the identifiers identify data entries in a protected memory; (10:31-35; 16:33-37)

wherein each of the memory protection keys comprises a modified version of a data sequence; wherein the modified version comprises a hash of the data sequence. (10:41-51; 17:1-30; 17:57-18:14)

Although England does not disclose rendering the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory protection key written to the volatile storage such that the memory protection key written to the volatile memory is inaccessible after completion of the memory command, the step of erasing sensitive information to prevent unauthorized disclosure protected information is well known in the art. Such a step prevents covert analysis of memory to determine the value of deallocated memory. For example, Bishop discloses a basic tenet of secure deletion of sensitive information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence, emphasis added) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made to render the memory protection key in the memory command inaccessible by overwriting at least a portion of the memory protection key written to the volatile memory is inaccessible after

completion of the memory command. One would be motivated to do so to securely remove sensitive information as known to one of ordinary skill. The aforementioned cover the limitations of claims 1 and 3-6.

(10) Response to Argument

On pgs. 8-11, Appellant argues that the combination of the Bryant reference with the Bishop reference does not render obvious the claimed invention. This argument is not persuasive because the secondary reference (Bishop) suggests the feature that Bryant fails to anticipate. The Bryant reference discloses an invention to provide protected access to pages of memory; when it is desirous to protect a page of memory by a program, a token is assigned to the portion of memory. The operating system defines a protection segment table (PVST) and protection verification table (PVPT), which maps token values to addresses in memory. The token value is also stored in a register, whereby the program can retrieve the token value and present it to the address translation and protection verification process to gain access to the protected page(s) of memory. See col. 5, line 55-col. 6, line 20. In particular, Bryant discloses:

> [i]n operation, the user program issues a special instruction that retrieves the previously stored token from its register. The user program then presents a virtual address of the datum in the program page which the instruction intends to alter and the token to an address translation and protection verification process. This process translates the virtual address into a real address in a manner that is common in the art ... On the other hand, if the real address is contained in a protected page, the process uses the PVST and PVPT to determine a token associated with the protected page. If this token matches the token provided by the special instruction, the instruction can alter the contents of the datum at the real address. However, if the tokens do not match, the operating system terminates the program for attempting an unauthorized memory access.

> In this manner, our token controlled page protection technique protects pages of memory from alternation by all programs not possessing an appropriate token. Col. 6, lines 1-20.

This portion of the Bryant reference clearly discloses a feature of presenting to the memory protection process a memory instruction that includes a virtual address of stored data and a token value, which was retrieved from a register. This step inherently requires storing the retrieved token value temporarily in hardware to perform the validation step. See fig. 3, reference no. 540. Furthermore, this temporary memory is separate from the aforementioned register and PVST/PVPT tables. Hence, it is with respect to this temporary memory by which the invention disclosed in Bryant anticipates the limitation "wherein the memory protection key in the memory command is written to a volatile memory." The feature defined in the claimed invention that is not anticipated by the Bryant reference is the step of "rendering the memory protection key in the memory write command that has been written to the volatile memory inaccessible by overwriting at least a portion of the memory protection key in the volatile memory upon completion of the memory write command." However, as outlined in the rejections above, the step of rendering sensitive information, which is stored in a particular memory, inaccessible by overwriting the information when the information is no longer needed is well known in the art as suggested by Bishop.

In view of the rejection based on the invention of Bryant as modified by the teaching of Bishop, Appellant argues that one of ordinary skill in the art would not be persuaded to render the temporary memory of Bryant inaccessible because Bryant expressly discloses storing the token in a register for future use by the authorized

program. See pgs. 9-10, Appeal Brief. Appellant's rationale stems from the notion that sensitive information should be rendered inaccessible only when sensitive information is no longer useful. However, the teaching expressed in Bishop is more nuanced than the one suggested by Appellant. In the computer arts, the usefulness of particular information depends on a specific instance of the information; i.e. a data value stored in a particular context. Hence, data representing sensitive information located in a specific volatile memory location is no longer useful when the *specific memory location* no longer requires use of the data. In view of this distinction, Appellant's argument that the token should not be erased because it is stored for future use pertains only to the data values stored in the register and the PVST/PVPT tables. In contrast, the data value representing the retrieved token, which is part of the special memory command and which is temporarily stored in volatile memory, is only used during the comparison step to validate the memory write command. Once the verification step is performed, the data value stored in this volatile memory is not used again and should be securely erased. Hence, contrary to Appellant's arguments, the teaching of Bryant as modified by Bishop suggests erasing the contents of this temporary memory to prevent invalid/improper use of the temporarily stored information. For this reason, it is respectfully submitted that Appellant's arguments against the combined teachings of Bryant and Bishop do not rebut the prima facie case of obviousness.

With respect to Appellant's arguments against the rejections in view of the combined teaching of Beukema and Bishop, Appellant merely alleges that "there is no

hint in Beukema of any desirability to render this protection key inaccessible by overwriting at least a portion of such protection key." Appeal Brief, pg. 12. Appellant's allegation is inadequate because it ignores the combined teachings of Beukema and Bishop. Beukema discloses a method and system to control access to the memory of a computer; a given memory region is protected by a corresponding key. When a user submits a work request to read from or write to a protected memory region, the user includes in the work request a key value corresponding to the key stored in a protection/translation table, which maps protection keys to memory regions. See paragraph 54. Clearly, in Beukema, the key values submitted in the form of a work request are sensitive because only a user with the proper key value can access protected memory. In addition, the Bishop reference provides the requisite motivation to render the protection key in the invention disclosed by Beukema inaccessible by overwriting the key. As outlined in the rejections above, Bishop discloses a basic tenet of secure deletion of sensitive information to prevent improper use of the information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. This teaching as applied to the invention of Beukema suggests the limitation of rendering a memory protection key in a memory command inaccessible by overwriting a portion of the memory key written to the volatile storage after the completion of the memory

command. For this reason, it is respectfully submitted that Appellant's arguments do not rebut the prima facie case of obviousness based on the combined teachings of Beukema and Bishop.

Appellant's arguments against the rejections based on Hind in view of Bryant and Bishop (see Appeal Brief, pgs. 13-16) are substantively similar to the arguments against the rejections based on the combined teaching of Bryant in view of Bishop; hence, it is respectfully submitted that Appellant's arguments do not rebut the prima facie case of obviousness based on Hind in view of Bryant and Bishop for the same reasons outlined above.

With respect to Appellant's arguments against the rejections based on England and Bishop, Appellant merely alleges that "England does not disclose or hint at rendering a memory protection key inaccessible by overwriting at least a portion of the memory protection key." Appeal Brief, pg. 16. Appellant's allegation is inadequate because it ignores the combined teachings of England and Bishop. England discloses a method and system to protect downloaded content stored in a key-secured storage area; the secured storage area is safeguarded such that only a trusted application can access the memory area. When an application requests access to content stored in the secured storage area, it provides an application storage key and a rights manager certificate to the operating system; the application is given access to the stored content when, *inter alia*, the storage key is validated by the OS and the certificate satisfies the

access predicate associated with the content. See col. 10, lines 26-53. Clearly, in England, the storage key value submitted in the request to access secured content is sensitive because only an application with the proper storage key value can access the stored content. In addition, the Bishop reference provides the requisite motivation to render the storage key in the invention disclosed by England inaccessible by overwriting the key. As outlined in the rejections above, Bishop discloses a basic tenet of secure deletion of sensitive information to prevent improper use of the information: "When a process finishes using a sensitive object (one that contains confidential information or one that should not be altered), the object should be erased, then deallocated or deleted. Any resources not needed should also be released." (pg. 901, last sentence-pg. 902, first sentence) Bishop further discloses an example of erasing sensitive information by overwriting the data. Pg. 902, 1st full paragraph. This teaching as applied to the invention of England suggests the limitation of rendering the protection storage key inaccessible by overwriting a portion of the storage key written to the volatile memory after the access request is processed. For this reason, it is respectfully submitted that Appellant's arguments do not rebut the prima facie case of obviousness based on the combined teachings of England and Bishop.

Appellant's arguments with respect to dependent claims 26-28 (see Appeal Brief, pg. 11) are moot in view of the new rejections. Note that dependent claims 26-28 were originally rejected in the final rejection mailed on 5/20/09 as incorrectly being dependent on independent claim 22. The instant rejections rectify this error; the rejections now

identify claims 26-28 as being dependent on claim 44. Claim 44 is a dependent claim of claim 25, which is a dependent claim of claim 22.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

This examiner's answer contains a new ground of rejection set forth in section **(9)** above. Accordingly, appellant must within **TWO MONTHS** from the date of this answer exercise one of the following two options to avoid *sua sponte* **dismissal of the appeal** as to the claims subject to the new ground of rejection:

(1) **Reopen prosecution.** Request that prosecution be reopened before the primary examiner by filing a reply under 37 CFR 1.111 with or without amendment, affidavit or other evidence. Any amendment, affidavit or other evidence must be relevant to the new grounds of rejection. A request that complies with 37 CFR 41.39(b)(1) will be entered and considered. Any request that prosecution be reopened will be treated as a request to withdraw the appeal.

(2) **Maintain appeal.** Request that the appeal be maintained by filing a reply brief as set forth in 37 CFR 41.41. Such a reply brief must address each new ground of rejection as set forth in 37 CFR 41.37(c)(1)(vii) and should be in compliance with the other requirements of 37 CFR 41.37(c). If a reply brief filed pursuant to 37 CFR

41.39(b)(2) is accompanied by any amendment, affidavit or other evidence, it shall be treated as a request that prosecution be reopened before the primary examiner under 37 CFR 41.39(b)(1).

Extensions of time under 37 CFR 1.136(a) are not applicable to the TWO MONTH time period set forth above. See 37 CFR 1.136(b) for extensions of time to reply for patent applications and 37 CFR 1.550(c) for extensions of time to reply for ex parte reexamination proceedings.

Respectfully submitted,

/Jung Kim/
Primary Examiner, Art Unit 2432

A Technology Center Director or designee must personally approve the new ground(s) of rejection set forth in section (9) above by signing below:

/Timothy P Callahan/
Director, Technology Center 2400

Conferees:

/Benjamin E Lanier/
Primary Examiner, Art Unit 2432

/Gilberto Barron Jr./
Supervisory Patent Examiner, Art Unit 2432